

(19) World Intellectual Property Organization  
International Bureau



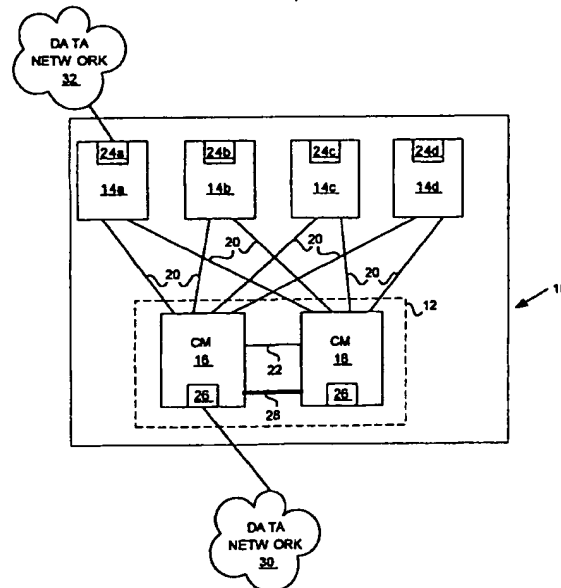
(43) International Publication Date  
3 May 2001 (03.05.2001)

PCT

(10) International Publication Number  
**WO 01/31848 A2**

- (51) International Patent Classification<sup>7</sup>: **H04L 12/24** (74) Agent: ZISCHKA, Matthew; Smart and Biggar, 438 University Avenue, Suite 1500, Box 111, Toronto, Ontario M5G 2K8 (CA).
- (21) International Application Number: PCT/CA00/01265
- (22) International Filing Date: 25 October 2000 (25.10.2000) (81) Designated State (national): CA.
- (25) Filing Language: English (84) Designated States (regional): European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE).
- (26) Publication Language: English
- (30) Priority Data: 09/427,683 27 October 1999 (27.10.1999) US  
Published:  
— Without international search report and to be republished upon receipt of that report.
- (71) Applicant: NORTEL NETWORKS LIMITED [CA/CA]; World Trade Center of Montreal, 380 St. Antoine Street West, 8th Floor, Montreal, Québec H2Y 3Y4 (CA).  
For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.
- (72) Inventor: SMITH, Kevin, W.; 1119 Grenon Avenue, Ottawa, Ontario K2B 1A9 (CA).

(54) Title: MANAGED NETWORK NODE INCLUDING MULTIPLE MANAGED RESOURCES AND METHOD



(57) Abstract: A network node including a control module and a plurality of resource modules is disclosed. Resources at the network node are managed using distributed software components. Management software resides at the resource modules, and preferably registers with a central agent at the control module that routes management requests to the individual resource modules. Management software may be distributed so that one resource module manages other modules. Legacy equipment may be managed by a management agent resident at another module. As well, resource modules may be managed by way of the central agent, or directly by way of ports on each resource module. Moreover, a second central agent on a redundant module may allow management at the node in the event an active central agent fails.



WO 01/31848 A2

MANAGED NETWORK NODE INCLUDING MULTIPLE MANAGED RESOURCES AND  
METHOD

FIELD OF THE INVENTION:

5

The present invention relates to networked computing devices and more particularly to a managed network node including a plurality of managed resources.

10

BACKGROUND OF THE INVENTION:

15

Network interconnected computing devices often allow for the remote management of the devices across the network. Physically distant devices can monitor and provision the operation of other devices.

20

Typically, network management information is stored at one or more managed devices interconnected with the network.

Management information may be stored within computer memory in a database known as a management information base ("MIB").

25

Software routines (often referred to as software objects) associated with the MIB are executed in response to modifying or querying the MIB. Interconnected devices may use software (referred to as a "manager agent") implementing known protocols to allow remote management of the network, typically by communicating with a software application at a managed device (referred to as a "management agent"). The management agent, in turn, accesses the MIB of the managed device. The MIB may store information about the names, status, and availability of network resources. The management agent may alter the MIB causing the managed device to change the availability and operation of existing network resources.

30

Many network devices, for example, include management agents that adhere to the Simple Network Management Protocol,

35

("SNMP") as detailed in the Internet Engineering Task Force, Request for Comments ("RFC"s) 1155, 1157, 1212, 1213, 1215, 1321, 1595, 1406, 1407 1450, 1573, 1595, 1695, 1901, 1902, 1903, 1904, 1905, 1906, 1907, 1908, 2328, 2358, 2271, 5 2272, 2273, 2274, 2275 the contents of all of which are hereby incorporated by reference. Others support a newer network management protocol as defined in the Common Management Information Protocol ("CMIP"), detailed in the International Telecommunication Union ("ITU") Recommendations X.710, X.711, 10 X.712, X.720, X.721, X.722, X.723, X.724, X.725, X.730, X.731, X.732, X.733, X.735, X.736, X.737, X.738, X.739, X.740, X.741, I.751, M.3100, and X.744, the contents of which are hereby incorporated herein by reference.

15 Further, many network devices, themselves include multiple resources. Some network devices, for example, include peripherals or multiple co-located processing modules. Network management of such devices is typically accomplished through use of a single MIB and a single associated management 20 agent located at each device. The MIB stores network management data relating to all interconnected resources at the device. This, however, creates the recognised problem that the MIB must be modified as resources are added or removed from the device.

25

U.S. Patent No. 5,678,006 accordingly suggests storing separate MIBs on each of a plurality of modules making up a network node. Each module additionally includes its own management agent, allowing management of an associated MIB. 30 Each management agent is accessed by way of a central network manager software component executing at the network node. This arrangement, however, appears to require that each managed module have the capacity to host a MIB and an associated management agent. This, in turn, excludes the 35 ability to include legacy equipment that does not have such a

capacity. Moreover, by requiring that distributed management agents be accessed through a central network manager software component, the arrangement lacks flexibility. Each module is dependent on the network manager.

5

Accordingly, an improved network node allowing network management of a plurality of interconnected resources in desirable.

10

#### SUMMARY OF THE INVENTION:

It is therefore an object of the present invention to provide a network node that allows flexible management of resources at the node.

15

In accordance with the present invention, resources at a network node are managed using distributed software components. Management software resides at resource modules, and preferably registers with a central agent at a control module that routes management requests to the individual resource modules. Management software may be distributed so that one resource module manages other modules. Legacy equipment may be managed by a management agent resident at another module. As well, resource modules may be managed centrally by way of the central agent, or independently by way of ports on each resource module. Moreover, a second central agent on a redundant module may allow management at the node in the event an active central agent fails.

20

25

30

In accordance with an aspect of the invention, a node for use within a communications network, includes a control module and a plurality of resource modules in communication with the control module, to provide data processing resources. The control module includes a port to receive management requests from the communications network and a processor operable to route the management requests to an identified one of the

resource modules based on identifying information associated with the management requests. Each of the modules, includes a processor operable to receive management requests from the control module and modify a management information base in response to the management requests and a port to receive management requests from a second network, and pass the management requests to an associated processor at the resource module to modify the management information base in response to the management requests.

- 10 In accordance with another aspect of the invention, a node for use within a communications network, includes a control module and a plurality of resource modules in communication with the control module, to provide data processing resources and adhering to a management protocol.
- 15 The control module includes a port to receive management requests from the communications network; routing data, identifying resources to be managed at the node, and corresponding local addresses; and a processor operable to route the management requests to an identified one of the
- 20 resource modules based on identifying information associated with the management requests using the routing data. Each of the resource modules, includes a processor operable to provide the control module with information to form the routing data and to receive management requests from the control module and
- 25 modify a management information base in response to the management requests.

- In accordance with another aspect of the invention, a method of routing network management requests at a network node including a control module and a plurality of managed
- 30 resources, includes forming routing information for routing received management requests, to manage the plurality of managed resources using data provided for the managed resources, at the control module; and routing the network management requests within the network node using the routing

information.

In accordance with another aspect of the invention, a resource module for use in a network node, includes a processor operable to manage resources at the resource module, and to provide a control module within the network node an indication of resources that may be managed at the resource module.

In accordance with another aspect of the invention, A resource module for use in a network node, includes a processor operable to manage resources at the resource module, and to provide a control module within the network node an indication of resources that may be managed at the resource module.

In accordance with another aspect of the invention, a computer readable medium, stores processor executable instructions, that when loaded at a resource module for use in a network node, adapt the resource module to manage resources at the resource module, and to provide a control module within the network node an indication of resources that may be managed at the resource module.

In accordance with another aspect of the invention, a computer readable medium, storing processor executable instructions, that when loaded at a network node including a control module and a plurality of managed resources, adapt the control module to: form routing information for routing received management requests, to manage the plurality of managed resources using data provided for the managed resources; and route the network management requests within the network node using the routing information.

In accordance with yet another aspect of the invention, a node for use within a communications network, includes first and second control modules and a plurality of resource modules

in communication with the first and second control module, to provide data processing resources at the node. Each resource module includes management software to manage resources at the network. The first control module includes a first port to  
5 receive management requests from the communications network and a processor operable to route management requests for one of the plurality of resource module based on identifying information associated with the management requests. The second control module includes a second port to receive  
10 management requests from the communications network and a processor operable to route management requests for one of the resource modules based on identifying information associated with the management requests, in the event of a switch of activity from the first control module to the second control  
15 module.

Other aspects and features of the present invention will become apparent to those of ordinary skill in the art, upon review of the following description of specific embodiments of  
20 the invention in conjunction with the accompanying figures.

#### BRIEF DESCRIPTION OF THE DRAWINGS:

In figures which illustrate, by way of example only,  
25 preferred embodiments of the invention,  
**FIG. 1** is a simplified block diagram of a network node, including control and resource modules exemplary of an embodiment of the present invention;  
**FIG. 2** is a simplified block diagram illustrating the  
30 organisation of software blocks on a portion of the node of **FIG. 1**;  
**FIG. 3** illustrates a routing table stored at the node of **FIG. 1**;  
**FIG. 4** illustrates steps performed by a resource module  
35 of **FIG. 1**, upon initialization;

**FIG. 5** illustrates steps performed by a control module of **FIG. 1**;

**FIG. 6** illustrates steps performed by a resource module of **FIG. 1**; and

5 **FIG. 7**, is a simplified block diagram of an additional configuration of the node of **FIG. 1**.

DETAILED DESCRIPTION:

10 **FIG. 1** illustrates, in simplified block diagram, a specific network node **10**, exemplary of an embodiment of the present invention. Network node **10** may be a node used to provide payload processing or routing services on a communications network, such as a data network or telephony  
15 network.

Network node **10** includes a control complex **12** and a plurality of resource modules ("RM"s) **14a**, **14b**, **14c** and **14d** (collectively and individually **14**). Four RMs **14a**, **14b**, **14c**  
20 and **14d** are specifically illustrated. Of course, node **10** could include any number of such RMs **14**.

Control complex **12** includes two control modules ("CM"s) **16** and **18**. Each CM **16**, **18** is physically interconnected to  
25 each of the plurality of RMs **14** by way of serial links **20**, as disclosed in U.S. Patent No. 5,842,007. Each of CM **16** and **18** includes its own processor, processor readable memory, and a switch fabric suitable to switch data between CMs **16** and **18** and RMs **14** in serial streams over serial links **20**, as for  
30 example detailed in U.S. Patent Application No. 08/721,095, the contents of which are hereby incorporated by reference. CMs **16** and **18** further preferably include their own ports **26** for connection to a data network.



5        RMs 14 and CMs 16 and 18 are preferably formed as separate cards mountable within a conventional rack-like housing, and interconnected by way of a back plane. Of course, RMs 14 and CMs 16 and 18 could otherwise be assembled to provide processing capability over several modules.

10        CMs 16 and 18 operate in redundancy, and are interconnected with each other by way of a cross-over link 22, and an inter-module communication link 28. In normal operation one CM 16 or 18 is active, while the other 18 or 16 is inactive, ready to become active in the event of a failure of the active CM. Cross-over link 22 signals a switch of activity from one CM 16, 18 to the other CM 18, 16. A switch of activity could be initiated by an active CM 16, or remotely by, for example, way of a management request. Preferably, module communication link 28 is a high capacity optical communication link, allowing direct exchange of data between CMs 16 and 18 in a serial stream using any conventional data exchange protocol.

25        Data switched between RMs 14 and CMs 16 and 18 is organised in frames with each frame preferably containing payload data and messaging data. As disclosed in U.S. Patent No. 5,842,007, pre-defined portions of each frame passed within the serial stream on serial links 20, are allocated for messaging data. Messaging data may be used to manage the operation of resources at each RM 14.

30        RMs 14 supply payload processing capabilities to node 10. The number and type of RMs 14 forming part of node 10 is application and provisioning dependent. Each RM 14 preferably has its own computing processing capability, and performs

local low-level control functions under supervision of one of CM 16 or CM 18. Each RM 14 may, for example, be an interface to a data or telephone network, or provide payload processing services such as pulse code modulated ("PCM") telephony services, voice over internet protocol telephony service, or the like. As such, each RM 14 typically includes one or more processors in communication with computer readable storage memory (not shown). Each RM may additionally be connected to a data network, such as second data network 30, by way of an additional physical port 24.

Preferably, node 10 is interconnected with one or more networks that may be used to exchange management data with node 10. Management data may be used to query the state of RMs 14; CMs 16 or 18; to provision resources at node 10; to provided software loads; or to monitor network or hardware resources or performance. In the embodiment of FIG. 1, CM 16 of node 10 is connected to first data network 30 by way of port 26, while RM 14a is connected to a second data network 32 by way of port 24a. First and second data networks 30 and 32 may be packet switched data networks adhering, for example, to the internet protocol ("IP"), or similar protocol. Data networks 30 and 32 could be interconnected and therefore effectively be part of a single network.

FIG. 2 is a simplified block diagram illustrating the organization of management software components at one RM 14a, and one CM 16. RMs 14b, 14c and 14d may include management software components similar to those at RM 14a. Similarly, CM 18 may include software components similar to those at CM 16.

In overview, and in a manner exemplary of the present invention, resources at a network node 10 are managed using

distributed software components. Exemplary RMs **14** may be centrally managed through control complex **12**, or independently managed through direct access through a network to each RM **14**.

5        Each RM **14** therefore preferably includes its own MIB **40**;  
its own software management software objects **33**; its own  
worker agent software **34**; network interface layer **36**; and port  
agent **38** software. Worker agent software **34** operates on MIB  
**40** and software objects **33** in a conventional manner in  
10    response to receiving network management request.

Network interface layer **36** receives management requests  
from an interconnected CM **16** (or **18**), by way of serial links  
**20** and passes these to worker agent software **34** to allow for  
15    central management of RMs **14**. Management requests are  
responded to by an associated worker agent **34**. Specifically,  
the worker agent **34** responds by updating or querying MIB **40** to  
manage the RM **14**, and sending suitable responses by way of  
network interface software **36** to an interconnected CM **16**.

20        Additionally, port agent software component **38** at each RM  
**14** may be in communication with an additional physical port,  
such as port **24** (**FIG. 1**) of an RM, to allow independent  
management of an RM **14** by way of an additional interconnected  
25    network, such as network **30**. Network management requests  
received by port agent **38** will be passed to worker agent **34**.

As will be appreciated, port agent **38**, worker agent **34**,  
network interface **36**, and software objects **33** may all be  
30    formed using traditional programming techniques known to those  
of ordinary skill in the art. Most preferably, port agent **38**,  
worker agent **34**, network interface **36**, and software objects **33**

are formed using a conventional object-oriented programming language such as C++. Resulting processor executable code is stored at each RM 14 and executed by a processor at the RM 14.

5 Data may be exchanged between software components in manners known to those of ordinary skill in the art.

A CM 16 similarly includes a network interface layer 42 software portion, to communicate by way of serial links 20 with interconnected RMs 14; its own worker agent 48 to modify  
10 its own MIB 50 and corresponding software objects 52; and a central agent 44 used to route management requests between CM 16 and RMs 14. An additional port agent 46 may receives management requests received by way of an external network 28 (FIG. 1). Again, each of these may be formed using  
15 programming techniques understood by those of ordinary skill in the art.

Network interface layers 36, 42 preferably allow for the exchange of management data, encapsulated in packets between  
20 RMs 14 and CM 16. Other data may also be exchanged using layers 36 and 42. Effectively, network interface layers 36 and 42 provide a network transport layer within node 10, thereby forming a local network among elements making up node 10. Packets are exchanged in the serial stream carried on  
25 links 20, in those portions reserved for management data. Network interface layers 36, 42 may for example exchange data using the internet protocol or any other suitable packet based protocol. For example, IPX, X.25 or a custom packetized protocol could be used. Network interface layers 36 and 42  
30 may accordingly include a suitable protocol stack. In the event IP is used, the protocol stack may include an IP stack supporting conventional internet protocols such as UDP/IP and TCP/IP protocols. In any event, packets containing management

requests and responses most preferably include source and destination addresses.

Each worker agent **34, 48** (at an RM and CM) and central agent **44** is, in turn, associated with a unique packet address, local to node **10**. In the event IP is used, each worker agent **34, 48** and central agent **44** may be identified by its own local IP address and port.

Network interface layer **42** at CM **16** may accordingly route management data to interconnected RMs **14** (and specifically identified worker agents **34**), by way of serial links **20**, identifying destination RMs **14** by way of destination addresses within packets destined for such RMs.

15

Port agent **46** at CM **16** receives management requests, and dispatches network management responses by way of network **28** interconnected to CM **16**. Port agent **46** may receive and dispatch management requests in any of a number of formats.

Port agent **46**, may for example receive management requests using SNMP v.3 messages, compliant with RFCs 2271, 2272, 2273, 2274 and 2275. That being so, port agent **46** may include a further TCP/IP stack allowing port agent **46** to communicate over network **28** using IP. In the event SNMP is used, management requests are included in SNMP protocol data units (PDUs), as for example detailed in RFC 2272.

Port agent **46**, parses the PDUs and provides management requests in some known protocol to central agent **44**. Therefore, if necessary, port agent **46** converts received management messages into the known protocol. For example, the known protocol used by central agent **44** may be the SNMP protocol (or some subset of this protocol). If port agent **46**

is designed to receive management requests in SNMP, no conversion may be necessary. On the other hand, if port agent 46 is adapted to receive requests using some other protocol, such as for example CMIP, port agent 46 will convert the received requests into the known SNMP protocol, in a conventional manner, and pass the converted requests to central agent 44. Of course, central agent 44 need not use the SNMP protocol, but could use any suitable management protocol or could use a custom protocol that may effect management of RMs 14 or CMs 16 or 18.

Central agent 44, in turn determines which of worker agents 34 or 48 any request should be passed to.

Central agent 44, includes routing information, preferably in the form of a look-up table 54 illustrated in FIG. 3, matching management request identifiers to worker agents. As illustrated, look-up table 54 preferably includes an SNMP objectID and a local network identifier for each resource forming part of node 10, that may be managed. Table 54 may also include an identifier of the physical slot of each RM 14 with which a resource is associated, and an SNMP ifIndex identifier for each resource. The network identifier preferably identifies a worker agent 34 at an RM 14 responsible for managing a particular resource by its local network address (ie. the address used by network interface layers 36, 42). As will become apparent table 54 may be a data structure stored within memory at CM 14 and dynamically formed each time node 10 is initialized. Moreover, each time a new RM 14 is added to node 10 or an RM 14 is removed, table 54 may be updated to reflect the added or removed resource.

Port agent 38 of RMs 14 similarly accepts network

management requests by way of network 30 (**FIG. 1**), provided by way of a physical network port 24 in communication with port agent 38, for independent management of resources at an RM 14. As each RM 14 does not include a central agent, management requests to an RM 14 port agent 38 are typically destined for a worker agent 34 associated with that particular RM 14. Moreover management requests for a particular RM 14 need not be in the format as those received at port agent 46 of CM 16. For example, port agent 46 of CM 16 may receive management requests in the CMIP format, while port agent 38 of each RM 14 may receive and process management requests in the hyper-text transfer protocol ("HTTP") format. Worker agents 34, on the other hand may modify associated MIBs in accordance with the SNMP protocol. Port agent 38 therefore preferably converts management requests received in a given protocol to the protocol/format used by worker agent 34 to manage RM 14. Thus, port agent 38 may include an HTTP web server, and appropriate HTTP to SNMP conversion software.

In operation, node 10 is first initialized. Steps S400 performed at each RM 14 are illustrated in **FIG. 4**. Specifically, operating system-like software (otherwise not illustrated) initializes each RM in step S402. Thereafter each worker agent 34 transmits an initialization message directed to the local network address of central agent 44 by way of network interface software 36 and link 20 to network interface 42 of CM 16, in step S404. The local network addresses of central agent 44, may be pre-programmed at each RM 14 or polled in step S402. The initialization message contains a list of network resources, preferably identified by their corresponding SNMP objectID variables at each RM. Additionally, a local network address identifying each worker

agent responsible for management of a resource is included in the initialization message. In the event IP is used by network interface layers **36** and **42**, this network address will be the IP address of worker agent **34**, which may be pre-

5 programmed as part of worker agent **34**. The initialization message may further include an SNMP ifIndex identifier, and slot number, identifying the physical location of a particular RM. As will be appreciated, the SNMP ifIndex identifier may be used to distinguish between two instances of the same

10 resource, as may for example be necessary in the presence of redundant resources.

Central agent **44**, in turn, updates table **54** (**FIG. 3**) to include the SNMP objectIDs of worker agents **34**, as well as

15 their network addresses, slot numbers of associated RM, and SNMP ifIndex numbers for each resource. Worker agents **34** are thus said to "register" their resources with central agent **44**. At the completion of the initialization, table **54** will thus identify all network resource at node **10**, along with the

20 network addresses of each worker agent **34** responsible for the management of a particular resource.

Any time a further RM is added to node **10**, initialization steps **S400** will typically be performed by the added RM. That

25 is, the added RM will typically also include software, such as the software of RM **14a** illustrated in **FIG. 2**, including its own series of software objects **33** that implement the management services provided by the new RM, in response to querying or modifying an associated MIB **40**. In addition, a

30 single worker agent **34** will typically be included as part of the software. This worker agent **34** represents all of the software objects **33** on the card. After being added to node **10**, the worker agent **34** initializes, by contacting central



agent 44, located at CM 16. This worker agent 34 "registers" software objects 33 that have been introduced on this RM, by providing a suitable registration message by way of network interface 36, link 20, and network interface 42. Optionally,  
5 the RM includes a port agent 38.

Additionally, a message, preferably in the form of an SNMP trap, indicating that an RM has been inserted may be delivered to identified manager agents awaiting messages from  
10 this new card, by way of port agent 38. The identity of the manager agents (typically in the format of a network address on network 32) may be maintained within non-volatile memory at the RM. Alternatively, the identity of manager agents could be obtained by an RM 14 or CM 16 or 18 using the conventional  
15 BOOTP protocol, detailed in RFCs 951, 1395, 1497, 1532, and 1542, the contents of all of which are hereby incorporated by reference.

After initialization, network management requests may be  
20 processed at CM 16 (FIG. 2) in steps S500 as illustrated in FIG. 5. Specifically, management requests may be centrally received at port agent 46 of CM 16 in step S502. For example, a computing device interconnected with network 30 executing a manager agent may dispatch the management request to port agent 46. Preferably, the management request is in  
25 the form of an SNMP message, identifying port agent 46 by its IP address. The request includes the SNMP objectID(s) of resources that are affected by this request. Port agent 46 relays the request to the central agent 44 in step S504. Here,  
30 the request will be queued in step S506 until central agent 44 may process the request in steps S508 and onward.

Central agent 44 relays the request to the worker agent

34 (or 48) that has registered responsibility for the particular objectID associated with the request in steps S508-S514. Specifically, central agent 44 uses table 54 to locate the local network address of the responsible worker agent in  
5 step S508.

In the event no worker agent has registered for the particular objectID, as determined in step S510, an SNMP error message is returned by central agent 44 to port agent 46 and  
10 then to network 28, in step S512.

In the event a worker agent 34 has registered, network interface software 40 forms at least one packet including the destination local address of the responsible worker agent, and  
15 the source local address identifying central agent 44. This packet(s) is dispatched by network interface 42, to the proper worker agent 34 (or 48) in step S514.

The packet may, for example, be broadcast to all RMs or  
20 only routed to the RM associated with the network address identifying the relevant worker agent 34. The recipient worker agent 34 uses the MIB 40 and associated software objects 33 at the RM to perform the requested management operations. The results are returned from the software objects  
25 33 / MIB 40 to the worker agent 34. In turn, worker agent 34 passes the results, along with the local network address of the central agent 44 to network interface 36.

Network interface 36 encapsulates the results in packets  
30 complying with the network protocol used by network interface 42 and network interface 36 having as source address, the local source address of worker agent 34 and destination address of central agent 44. Network interface 42, receives

the response and passes its contents to central agent 44 in step S512. Central agent 44 passes the response to port agent 46, where the response is converted to the format/protocol used by port agent 46. As well, a response including packets  
5 compliant with network 28 including the source network address of port agent 46 and destination address of the manager agent is formed. The response is passed by way of network 28 to the manager agent originating the request in step S518.

10 Management of resources at CM 16 may similarly be effected through worker agent 48 and MIB 50 and software objects 52. Management requests are received at port agent 46. Worker agent 48 registers its resources in a manner similar to worker agents 34. Central agent 44 may communicate  
15 with worker agent 48, either directly or through network interface 42 (as illustrated in FIG. 2).

Additionally, resources at each RM may be independently managed by way of a local port agent 38. As illustrated in  
20 steps S600 of FIG. 6, port agent 38 of an RM may receive a management request from a manager agent by way of data network 32 in step S602. This management request may be received in a suitable protocol supported by port agent 38. Port agent 38, in turn if necessary converts the management request to the  
25 management format supported by worker agent 38, and passes the request to worker agent 34 in step S604. If worker agent 34 is unable to process the request, an error message is returned to port agent 38, and in turn to the management agent originating the request in step S606. Otherwise, the  
30 management request is processed by worker agent 34 in step S608 which in turn queries or modifies MIB 40 and associated software objects 33. A resulting response is passed to port

agent **38**, where it is converted into the format expected by the manager agent originating the management request in step **S610**. Port agent **38**, thereafter passes the response as converted to the manager agent in step **S612**.

5

As should now be appreciated, different RMs **14**, and services provided at these RMs **14** may utilize different, incompatible operational procedures. New services composed of new hardware and new software can be added without disrupting existing services. MIBs, worker agents and software objects for new hardware and software can be added without requiring manual update of central agent **44** or the MIB **50** at CM **16**. Port agents **38**, **46** receive management requests in a desired format/protocol, and preferably convert these request to a standardized format/protocol understood by worker agents **34**. Worker agents **34** at the RMs preferably receive requests using the standardized format/protocol, and process requests to manage the resources. Conveniently, in a preferred embodiment, services may be managed through the port agents **38** local to each RM. Central management of a system may also be effected through the port agent **46** of the CM **16**.

As well, RMs **14** may operate in redundancy, with one redundant RM prepared to assume the role of an operating RM, as for example detailed in U.S. Patent Application No. 08/721,095. Worker agents **34** and managed resources can also act as backups of one another, with one worker agent **34** on an inactive RM, ready to assume the role of a worker agent **34** on an active RM in the event of a failure. Should this be the case, the default address within table **54** for any resource identified by its objectID identifier will identify a resource on an active RM, and managed by a worker agent **34** at an active RM. In the event an active RM fails, operating system

software at system **10** may notify central agent **44**, which in turn may update table **54**, so that entries for the previously active resources may be deleted. If redundant resources having the same objectID identifier are still at system **10**,  
5 central agent **44** may route future management requests to these resources and an associated worker agent **34** using table **54**. As should therefore be appreciated, active and redundant resources may be managed by different worker agents **34**. Alternatively, a specific resource (redundant or active) may  
10 be addressed by a manager agent using the SNMP ifIndex identifier for the resource. In the event an active RM having an active worker agent **34** fails, network interface software **42** may route messages for the formerly active worker agent **34** to the redundant worker agent.

15

Similarly, when an RM is removed from node **10**, the central agent **44** is preferably made aware of such a removal by way of an event message from the operating system software at node **10**. For example, the operating system software may  
20 indicate that an RM has been removed from a particular slot. Central agent **44**, in response, modifies table **54** and may notify any manager agent that a card has been removed. Central agent **44** may, for example, delete all entries of table **54** identifying resources at a particular slot. Moreover,  
25 central agent **44** will no longer responds to management requests to access software objects which reside on that RM.

Legacy equipment may be managed using a configuration of node **10** illustrated in **FIG. 7**. In this arrangement node **10** is  
30 configured in a manner substantially identical to that illustrate in **FIG. 1**. Node **10** of **FIG. 7**, however includes five RMs (only RM **14a**, **14d** and **14e** are illustrated. One or more RMs **14e** in communication with CM **16** (and CM **18** - not

shown) does not include a MIB or a worker agent. The remaining RMs may include management software as illustrated in FIG. 2. RM 14e, however, may not have the capacity to host a worker agent or MIB. Instead, several worker agents 34 and 5 34b and possibly several MIBs 40 and 40b may be loaded on a particular RM 14d that is used to manage RM 14e.

Further, several software objects 33b used to manage resources at RM 14e may be loaded at RM 14d. At 10 initialisation of node 10, each additional worker agent 34b at RM 14d registers with central agent 44, to provide identifiers of added worker agents and of the resources managed by that worker agent 34b. In this case, resources managed by worker agent 34b are actually located at RM 14e. Central agent 44, 15 in turn updates the contents of table 54 (FIG. 3) to reflect that any management requests received at port agent 46 may be routed to the appropriate worker agent 34b at RM 14d. Now, any management requests for RM 14e will be routed and processed by the worker agent 34b at RM 14d. Software objects 20 33b, in turn modify or monitor resources at RM 14e, by way of the back plane interconnecting RMs or by using a legacy protocol that may be peculiar to RM 14e. Similarly, worker agent 34b may also be accessed by way of port agent 38.

25 As should be appreciated, any number of worker agents and MIBs could be loaded onto any particular RM, thus allowing the distributed management of multiple other RMs and their resources. For example, three or more worker agents and associated MIBs could be loaded at a particular RM, to allow 30 management of two or more additional legacy RMs.

Clearly, the central agent 44 on CM 16 is a potential single-point-of-failure. Accordingly, in the embodiment of

**FIG. 1**, central agent **44** is replicated on redundant CM **18**. All central agent data, including table **54** (**FIG. 3**) and the contents of MIB **50**, is therefore replicated at the redundant CM so that the "spare" processor can assume activity if a failure occurs on the active processor. If desired, redundancy for worker agents and software objects may be similarly supplied.

Central agent **44** at each of CM **16** and **18** should  
10 synchronize data to the backup CM using a variety of traffic paths on node **10**. As such CMs **16** and **18** may also execute replication software used to update the MIB of the redundant CM. The replication software could have its own local network address. Thus, replication data may be distributed by way of  
15 network interface software **42** at both CMs or alternatively by way of link **28**.

During normal operations, the central agent **44** on CM **16** notifies the central agent on CM **18** of any changes to its  
20 internal data. In this fashion, both central agents **44** of CMs **16** and **18** will maintain identical management data. In the event of a switch of activity from CM **16** to CM **18**, as indicated by a switch of activity signal on connection **22**. In response the otherwise inactive redundant CM **18** becomes  
25 active, and may notify interconnected manager agents of the change in activity. Manager agents, may in turn contact the port agent of CM **18** with future management requests. Alternatively, hardware at CMs **16** and **18** may cause the port **26** of CM **18** to assume the same network address in response to a  
30 switch in activity. Manager agents would therefore not need to be made aware of the switch of activity.

As will be appreciated, many different possible

configurations of worker, port, and central agents even for the identical hardware environment are possible.

Moreover, while node 10 has been described as supporting  
5 SNMP v.3, a person skilled in the art will easily appreciate that node 10 could easily be modified to support another network management protocol, in a manner exemplary of the present invention. Node 10, could for example support SNMP, SNMP v.2, CMIP or similar management protocols.

10

As well, while the software blocks, data and data structures have been illustrated as clearly delineated, a person skilled in the art will appreciate that the delineation between blocks and data is somewhat arbitrary. Numerous other  
15 arrangements of software blocks and data are possible. Moreover, software adapting the various components of node 10 to function in manners exemplary of the present invention may be distributed in whole, or in modules, by way of a computer readable medium, such as a diskette, CD-ROM, EPROM, tape or  
20 the like.

The above described embodiments, are intended to be illustrative only and in no way limiting. The described embodiments are susceptible to many modifications of form,  
25 arrangement of parts, and details and order of operation. The invention, rather, is intended to encompass all such modification within its scope, as defined by the claims.



## WHAT IS CLAIMED IS:

1. A node for use within a communications network, said node comprising:
  - a control module;
  - a plurality of resource modules in communication with said control module, to provide data processing resources;
  - said control module comprising a port to receive management requests from said communications network and a processor operable to route said management requests to an identified one of said resource modules based on identifying information associated with said management requests;
  - each of said resource modules, comprising
    - a processor operable to receive management requests from said control module and modify a management information base in response to said management requests;
    - a port to receive management requests from a second network, and pass said management requests to an associated processor at said resource module to modify said management information base in response to said management requests.
2. The node of claim 1, wherein said management requests from said communications network form part of simple network management protocol (SNMP) requests.
3. The node of claim 3, wherein said management requests from said second network are provided using a hypertext transfer

protocol (HTTP).

4. The node of claim 1, further comprising a routing table used by said control module to route said management requests.
5. The node of claim 1, further comprising a network interface at each of said resource modules, and a network interface at said control module to route said management requests.
6. A node for use within a communications network, said node comprising:

a control module;

a plurality of resource modules in communication with said control module, to provide data processing resources and adhering to a management protocol;

said control module comprising

a port to receive management requests from said communications network;

routing data, identifying resources to be managed at said node, and corresponding local addresses;

a processor operable to route said management requests to an identified one of said resource modules based on identifying information associated with said management requests using said routing data;

each of said resource modules, comprising a processor operable to provide said control module with information to form said routing data and to receive management requests from said control module and modify a management information base in response to said management requests.

7. The node of claim 6, wherein said resource modules adhere to the simple network management protocol (SNMP).
8. The node of claim 7, wherein said information includes an SNMP objectID parameter for resources to be managed by said resource module.
9. The node of claim 8, wherein said information includes an address identifying said resource module at said node.
10. A node for use within a communications network, said node comprising:
  - a control module;
  - first and second resource modules in communication with said control module, to provide data processing resources at said node;
  - said first resource module comprising management software to manage resources at said first and second resource modules;
  - said control module comprising a port to receive management requests from said communications network and a processor operable to route management requests for said first and second modules to said first resource module based on identifying information associated with said management requests;
  - said first resource module comprising a processor operable to receive management requests from said control module and manage said first and second resource modules in response to said management requests.
11. The node of claim 10,

wherein said first resource module comprises first and second management information bases for storing management information relevant to resources at said first and second resource modules, respectively,

and wherein said first resource module processor is further operable to modify said first and second management information bases to manage resources at said first and second resource modules, respectively.

12. A method of routing network management requests at a network node including a control module and a plurality of managed resources, said method comprising:

at said control module, forming routing information for routing received management requests, to manage said plurality of managed resources using data provided for said managed resources; and

routing said network management requests within said network node using said routing information.

13. The method of claim 12, wherein said data includes an identifier of each of said managed resources, usable by a network management protocol.
14. The method of claim 13, wherein said data further comprises an identifier, identifying management software at said node operable to manage each of said managed resources.
15. The method of claim 12, wherein said forming further comprises forming a table including management request identifiers, and identifiers of each of said resources at said node.
16. A resource module for use in a network node, comprising:

a processor operable to manage resources at said resource module, and to provide a control module within said network node an indication of resources that may be managed at said resource module.

17. The resource module of claim 16, further comprising a network interface to provide said indication of resources to said control module.

18. The resource module of claim 16, wherein said processor is operable to manage resources at said resource module using a simple network management protocol (SNMP).

19. Computer readable medium, storing processor executable instructions, that when loaded at a resource module for use in a network node, adapt said resource module to manage resources at said resource module, and to provide a control module within said network node an indication of resources that may be managed at said resource module.

20. Computer readable medium, storing processor executable instructions, that when loaded at a network node including a control module and a plurality of managed resources, adapt said control module to:

form routing information for routing received management requests, to manage said plurality of managed resources using data provided for said managed resources; and

route said network management requests within said network node using said routing information.

21. A node for use within a communications network, said node comprising:

first and second control modules;

a plurality of resource modules in communication with said first and second control module, to provide data processing resources at said node, each including management software to manage resources at said network

said first control module comprising a first port to receive management requests from said communications network and a processor operable to route management requests for one of said plurality of resource module based on identifying information associated with said management requests;

said second control module comprising a second port to receive management requests from said communications network and a processor operable to route management requests for one of said plurality of resource modules based on identifying information associated with said management requests, in the event of a switch of activity from said first control module to said second control module.

22. The node of claim 21, wherein said switch of activity is signalled by said first control module to said second control module.
23. The node of claim 22, wherein said switch of activity is signalled as a result of a failure at said first control module.
24. The node of claim 21, wherein each of said first and second control modules stores routing information used to route said management requests among said plurality of resource modules.
25. The node of claim 22, wherein said routing information at said first and second control module is periodically

synchronised.

26. The node of claim 24, wherein data to synchronise said routing information at said first and second control module is provided by said first control module to said second control module.

1/6

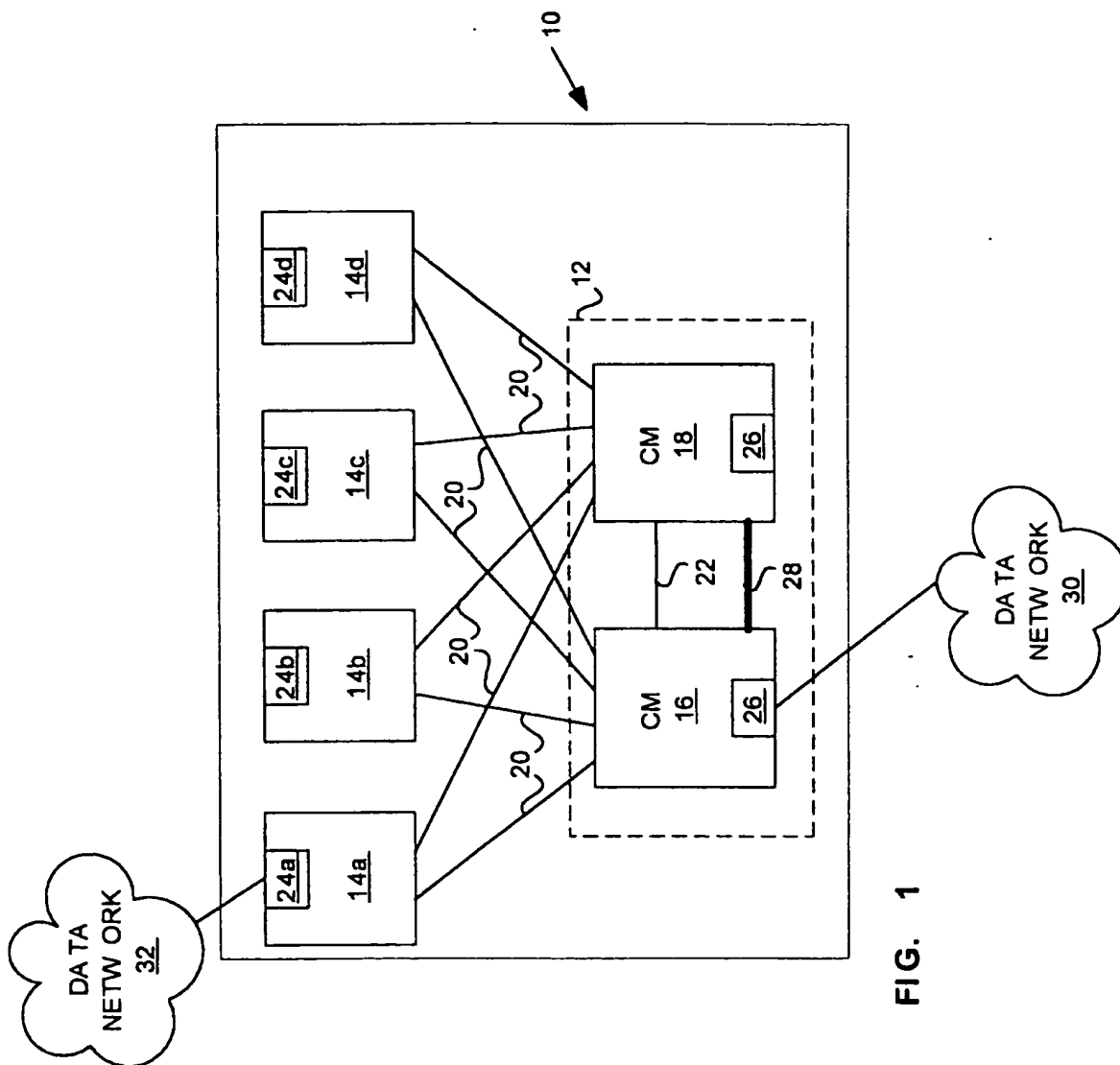
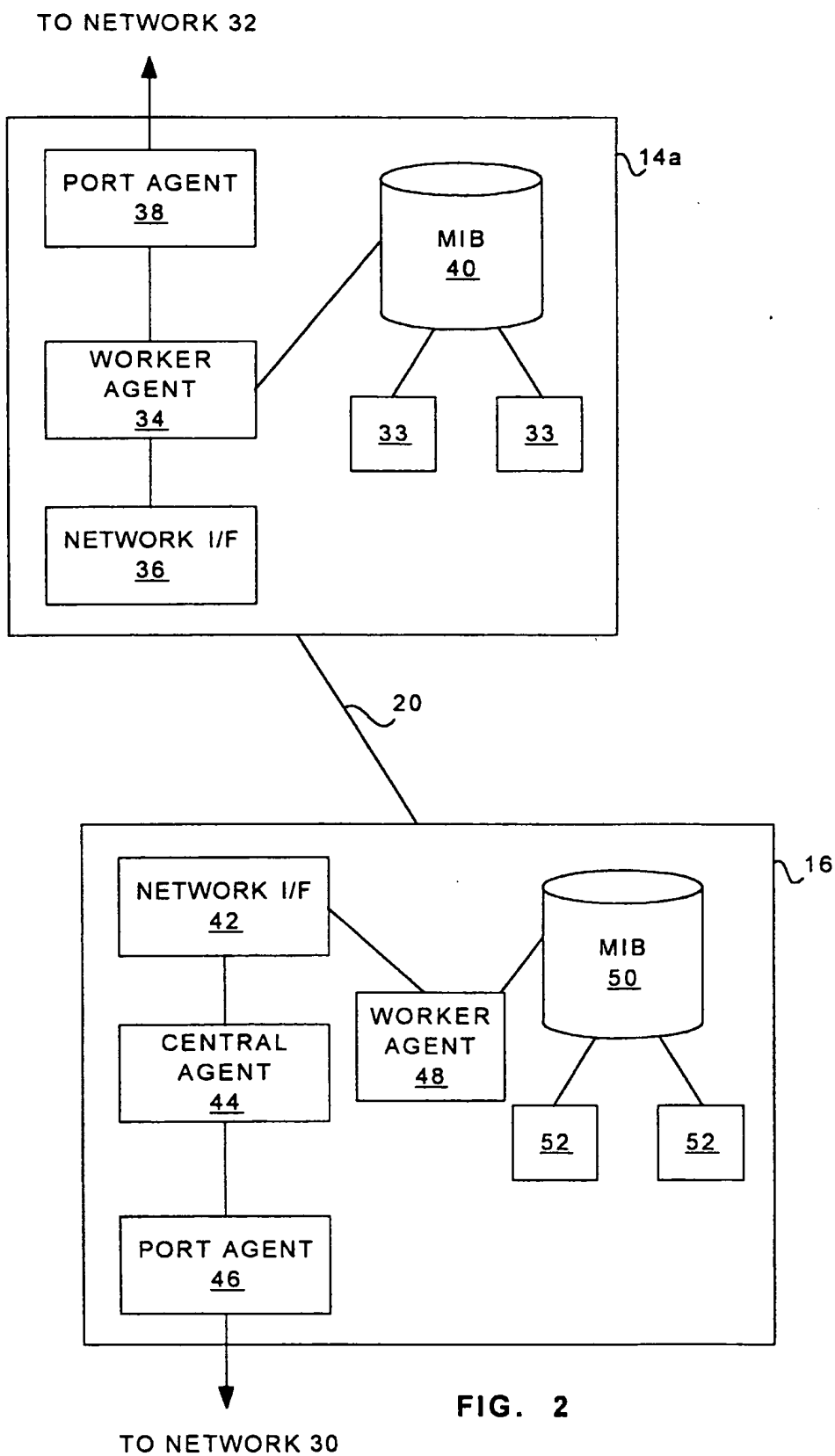


FIG. 1



2/6



OBJECT ID	NETWORK ADDRESS	SLOT	ifIndex

54

FIG. 3

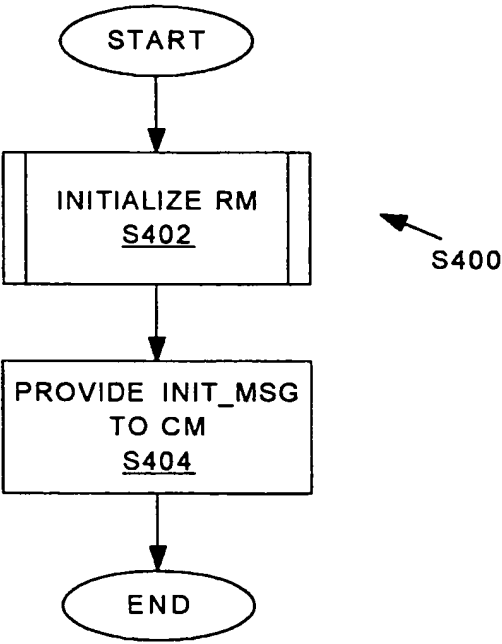


FIG. 4

4/6

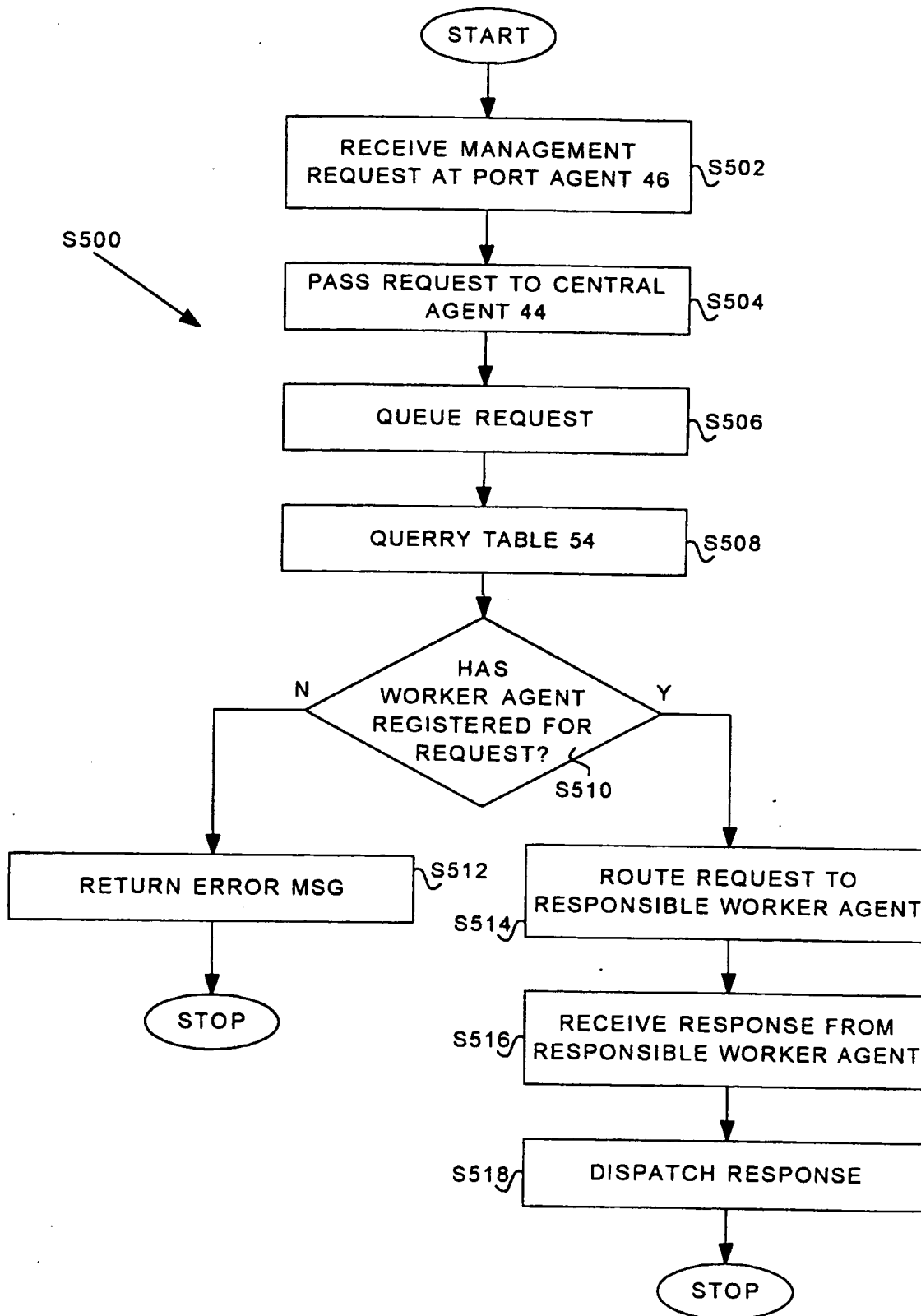


FIG. 5

5/6

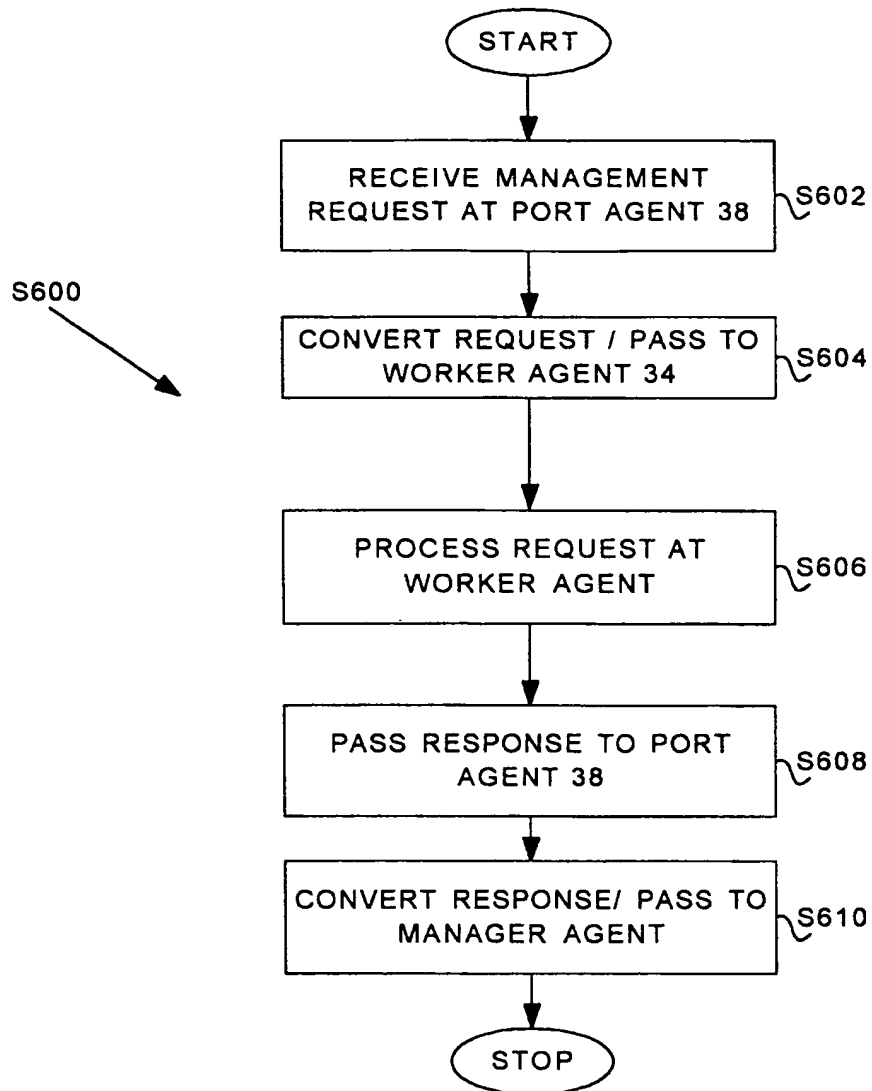


FIG. 6

6/6

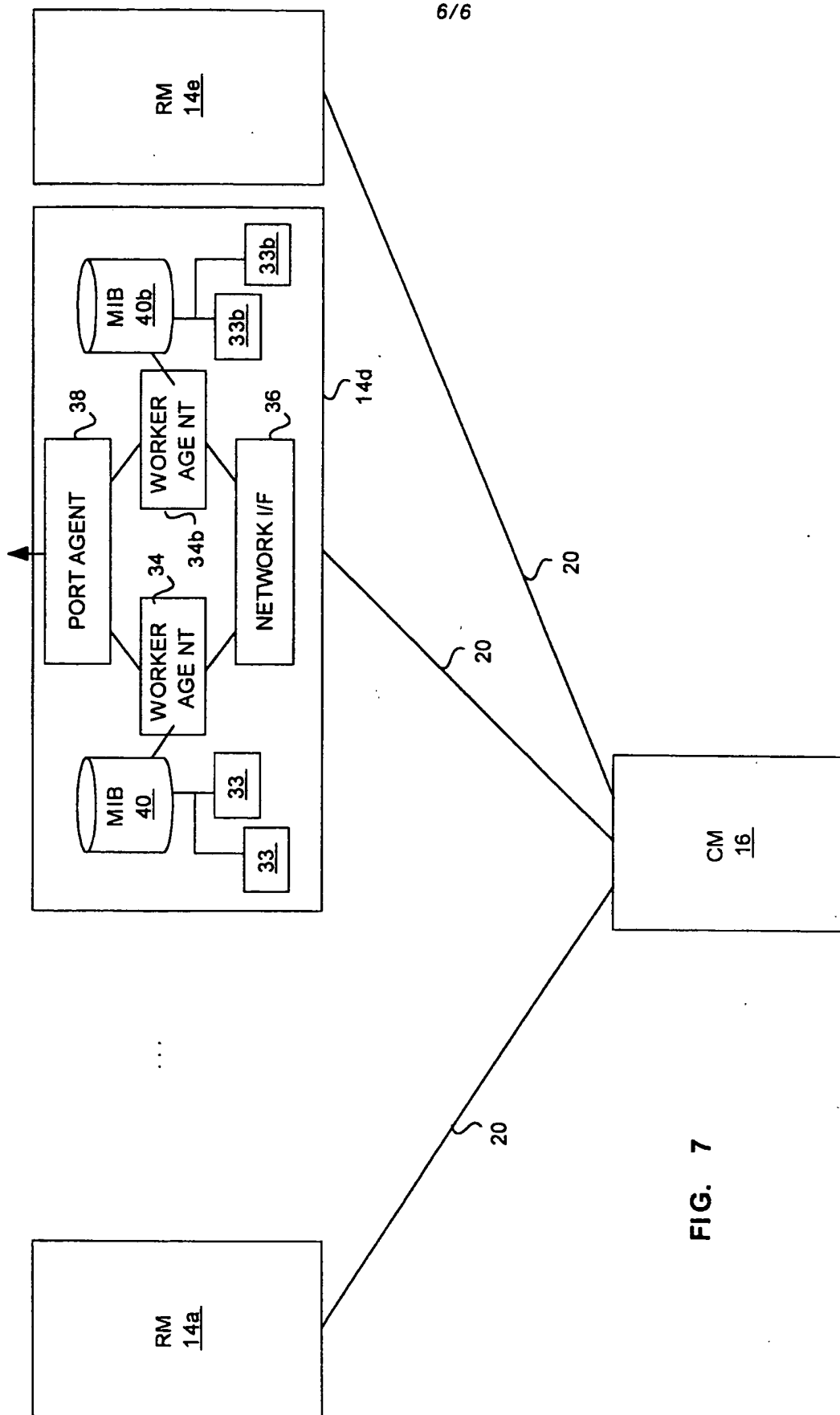


FIG. 7

**THIS PAGE BLANK (USPTO)**